RECEIVED
OCT 23 1974

Req Copy

Postel

RFC #642

NIC 30872

Ready Line Philosophy and Implementation

Jerry Burchfiel

5 July, 1974

I. Introduction

BBN Report #1822, Specifications for the Interconnection of a Host and an IMP, gives a complete specification of the Host-IMP interface. However, the authors of this document bent over backward to avoid issuing arbitrary dictatorial directives to host interface implementers. They succeeded admirably in this goal by describing the IMP implementation, and suggesting similar behavior on the part of the host.

ARPA has appointed a PDP-11 local host interface standardization committee composed of myself, Dave Retz of SCRL, and Yuval Peduel of MIT Lincoln Labs. During our review of various interfaces designed by the ARPA community, we have found total chaos, confusion, and misunderstanding about the recommended host interface implementation.

This note is an attempt to make explicit the recommendations which are implicit in Report #1822. It provides a cookbook for interface implementers, including a set of recommended do's and don't's in the common problem areas. This document has been reviewed and approved by the BBN IMP group.

II. Ready-line Philosophy

The following is an attempt to spell out in detail a consistent plan for operation of the IMP ready line and host ready line with the following objectives:

1. Reliably resynchronize and resume transmission after a temporary lapse of service and possible loss of state information by either system.

2. Make the programming of the host interface as simple as possible. This will minimize bugs, and make it possible to create a small ROM network_bootstrap loader.

First, consider the IMP ready line. When it drops, the IMP has suffered a possible loss of state, so the message in transit from the IMP to the host (if any) is likely to be incomplete. Similarly, the message in transit from the host to the IMP (if any) is likely to be incomplete. Both the host and the IMP must recognize this explicitly by sending a message intended to be thrown away* (which may be appended to the current message) and throw away the message currently being received. (Both the host - IMP message and the IMP - host message).

The simplest arrangement for the host's interface driver is a pair of processes, one sending messages and the other receiving messages. This drop of the IMP's ready line must be provided as an error status bit to each process. However, the two processes will need to clear this condition independently: the simplest implementation is an Input Error flop and an Output Error flop. Both flops are set by a drop of the IMP's ready line, and they are cleared independently under program control.

When the IMP raises its ready line, each contact bounce will again set the Error flops in the host's interface. To insure that messages are not flowing across the interface at this time, assertions of the signals "there's your IMP bit" and "ready for next host bit" have been delayed sufficiently in the IMP to guarantee that the IMP ready line has stabilized.

- - - - - - - - - -

*The standard convention uses the host-IMP NOP message.

III. Programming

The interface driver processes can be described simply:

```
INPUT:  Wait until an input buffer is available
        Wait until IMP ready
        Start input
        Wait until input is complete
        IF Input Error
        THEN clear Input Error // Flush smashed message.  Input
                               // buffer will be reused.
        ELSE queue message on input queue
        GOTO INPUT

OUTPUT: Wait until a message is present on output queue
        Wait until IMP ready
        Start output
        Wait until output is complete
        IF Output Error
        THEN clear Output Error // Smashed message is flushed
        ELSE deque message from output queue // Free up
                                             // output buffer
        GOTO OUTPUT
```

The only initialization required for system startup or restart is clearing the host READY flop, waiting 1/2 second, and setting the host READY flop. Simply starting (or restarting) the above processes will properly resynchronize host-IMP communication. As explained in RFC #636, the IMP ready line (and error flops) should only affect the two processes above: this resynchronization should be invisible to the NCP, and should have no effect on the connection data base. The NCP will be resynchronized or reinitialized by the type 10 IMP-to-host message "interface was reset."

Actually, it is possible to share a single Error flop between the input and output processes by implementing Input Error and Output Error as software flags. A process testing for error must test both the Error flop and its own flag. An interlock is required (e.g. a mutual exclusion semaphore) to guarantee that only one process at a time is testing and modifying the flags. If the Error flop is set, the process must copy it into the other process' flag before clearing the flop and its own flag.

IV. Host Ready Line Implementation

When the host drops and raises its ready line, the IMP behaves in a fashion symmetric to that outlined above. Of course, this drop indicates that the state of the host's interface driver, as well as the current incoming and outgoing messages, are likely to be lost. The appropriate action is triggered by setting the Error flop or flops in the host interface, and the processes specified above will correctly resynchronize message flow in both directions. Of course, to guarantee that messages are not flowing across the interface while the host ready line is undergoing contact bounce, the host must delay transmission until its ready line has stabilized. This may be done in two ways:

Hardware: an integrating one-shot driven by the host ready line flop is ANDed with "there's your host bit" and "ready for next IMP bit" to guarantee that message transfer will not start until the ready flop has been on for 1/2 second.

Software: the initialization program executes a 1/2 second wait after setting the host ready flop before permitting input or output to begin.

V. Summary

This determines the specification READY line controls for the host's interface to the IMP:

1. It contains a program settable/clearable host READY flop which drives a relay closure to the IMP.

2. It detects the IMP's ready signal as a program-readable status condition. (But not an interrupt condition)

3. It contains one or two ERROR flops set when either the host READY flop is off or the IMP ready signal is off. The flop (flops) is a program-readable and program-clearable status condition. (But not an interrupt condition). These status flops must not be cleared by system initialization.

4. If hardware stabilization of the host's READY line is provided, it is a 1/2 second integrating one-shot driven by the host READY flop. This signal is ANDed with "there's your host bit" and "ready for next IMP bit".